IN THE CLAIMS:

1.   (currently amended) A method of providing a fair exchange of messages to players of a distributed real-time multi-player game taking place over a communications network, said method comprising the steps of:

sending update messages generated by a game server toward said players, each update message having a respective update message number associated therewith;

receiving action messages from said players, wherein each action message received from a player comprises an indication of an update message with which the action message is associated and a reaction time associated with the action message, said reaction time being a difference between a reception time of the update message received for the player and a sending time of the action message sent by the player in response to the update message;

computing, for each received action message, a respective delivery time for use in delivering the action message for processing by the game server, wherein one of a plurality of delivery time ~~formula~~ formulas is utilized for the action message ~~is utilized~~ depending on whether the action messages arrive in order and whether the action messages arrive within their wait timeout periods; the plurality of delivery time ~~formula~~ formulas comprising a first delivery time formula utilized when the action messages arrive in order and within their wait timeout periods, a second delivery time formula utilized when the action messages arrive out of order but within their wait timeout periods, and a third delivery time formula utilized when the action messages arrive outside their wait timeout periods; wherein a wait timeout period for a player is calculated based on an expected round trip time between a game server proxy and a player ~~proxy;~~ proxy, wherein the game server proxy is operable in connection with said game ~~server,~~ server and the player proxy is operable in connection with said ~~player.~~ player;

queuing, in real-time, each received action message for use in delivering the action message for processing by the game server, wherein the queued action messages are arranged in an order of increasing update message number and are further arranged for each update message in an order of increasing reaction time; and

delivering, in real-time, said queued action messages for processing by said game server.

2.   (previously presented) The method of claim 1, wherein said game server proxy is operable in connection with said game server for receiving said action messages, computing said delivery times for said action messages, queuing said action messages, and delivering said action message to said game server.

3.   (previously presented) The method of claim 2, wherein each action message received at said game server proxy is delayed until its computed delivery time is reached to ensure fair processing of the action messages sent from all players.

4.   (previously presented) The method of claim 2, wherein, for each update message, said game server proxy associates the update message number with the update message for tracking an update message to which an action message responds.

5.   (previously presented) The method of claim 2, wherein, for each update message, said game server proxy records a sending time for said update message and associates said sending time with said update message.

6.   (previously presented) The method of claim 1, wherein said player proxy is operable in connection with said game server for receiving said update messages from said game server and forwarding said update messages to said game players, and for receiving said action messages from said game players and forwarding said action messages to said game server.

7.   (previously presented) The method of claim 6, wherein said player proxy records said reception time of an update message and uses said reception time to calculate said reaction time once said action message is sent by said player.

8.   (previously presented) The method of claim 6, wherein, for each action message, said player proxy sends with said action message an update message number of an update message with which said action message is associated, a reaction time of said action message, and an action message number of said action message.

9.   (previously presented) The method of claim 6, wherein a message split mechanism is employed at said player proxy when multiple update messages are outstanding, wherein said message split mechanism associates an action message with a window of update messages, wherein said message split mechanism calculates a respective reaction time for each outstanding update message in said window of update messages.

10-11. (cancelled)

12.   (previously presented) The method of claim 2, wherein said game server proxy, when an action message is received, computes said delivery time for said action message and computes a position in a queue where said action message should be inserted.

13.   (cancelled)

14.   (previously presented) The method of claim 1, wherein the delivery time of an action message is calculated before the action message is queued, and recalculated upon new action message arrival when action messages arrive in order or out of order but within their wait timeout periods.

15.   (previously presented) The method of claim 1, wherein each of the action messages has a respective action message number associated therewith; wherein, when action messages arrive at said game server proxy out of order, the action message numbers are used by said game server proxy to order action messages from a specific player and to determine whether all earlier action messages sent by said specific player have arrived.

16.   (previously presented) The method of claim 1, wherein the delivery time of an action message is calculated before the action message is queued, and recalculated upon new action message arrival and action message delivery when action messages arrive outside of the wait timeout period.

17. (previously presented) The method of claim 6, wherein, when an action message is sent by a player for multiple update messages, a set of tuples is tagged to the action message by the player proxy, wherein the set of tuples includes multiple reaction times associated with the respective multiple update messages for which the action message is sent.

18. (previously presented) The method of claim 1, wherein a window of update messages for which action messages are needed is indicated by the game server proxy to the player proxy, the window being based on the determination by the game server proxy about when to stop accepting action messages corresponding to a particular update message.

19-21. (cancelled)

22. (currently amended) A system for a distributed real-time multi-player game, said system providing a fair exchange of messages to players of the distributed real-time multi-player game taking place over a communications network, said system comprising:

a game server for sending update messages to said players and receiving action messages from said players;

wherein each update message has a respective update message number associated therewith;

wherein each action message comprises a reaction time associated with the action message, said reaction time being a difference between a reception time of the update message received for the player and a sending time of the action message sent by the player in response to the update message; and

a server proxy for delivering said action messages for processing by said game server in an order of increasing reaction time, said server proxy adapted for:

receiving said action messages from said players;

computing, for each received action message, a respective delivery time for use in delivering the action message for processing by the game server, wherein one of a plurality of delivery time ~~formula~~ formulas is utilized for the action message ~~is utilized~~

depending on whether the action messages arrive in order and whether the action messages arrive within their wait timeout periods; the plurality of delivery time ~~formula~~ formulas comprising a first delivery time formula utilized when the action messages arrive in order and within their wait timeout periods, a second delivery time formula utilized when the action messages arrive out of order but within their wait timeout periods, and a third delivery time formula utilized when the action messages arrive outside their wait timeout periods; wherein a wait timeout period for a player is calculated based on an expected round trip time between a game server proxy and a player ~~proxy;~~ proxy, wherein the game server proxy is operable in connection with said game ~~server,~~ server and the player proxy is operable in connection with said ~~player.~~ player;

queuing, in real-time, each received action message for use in delivering the action message for processing by the game server, wherein the queued action messages are arranged in an order of increasing update message number and are further arranged for each update message in an order of increasing reaction time; and

delivering, in real-time, said queued action messages for processing by said game server.

23.   (previously presented) The system of claim 22, further including a plurality of player proxies, each of said player proxies adapted to:

receive said update message from said game server at said player proxy;

record said reception time of said update message at said player proxy; and

calculate said reaction time using said reception time, said reaction time transmitted by said player proxy in connection with said action message.

24.   (previously presented) The system of claim 23, wherein each of said player proxies is adapted to send said update message number, said reaction time and an action message number with said action message.

25.   (previously presented) The system of claim 23, wherein said reaction time is used by said multi-player game at said server proxy to order responses by said players to thereby provide said fair exchange of messages.

26.   (cancelled)

27.   (previously presented) The system of claim 22, wherein said server proxy, when said action message is received, computes a position in a queue where said action message should be inserted and a local delivery time at which said action message is to be delivered to said game server.

28-29. (cancelled)

1009313-1